

... "RECITAR CANTANDO" ...

...CANTARCI RECANDO
CANTANDO TARRECI
RETANDO CATARCI
RECINTA DOTAR
CANDORE CITAR
CANTAN DOCI
CANTANDO RETAR
CI RECAN TARDO
CANTAR RECIDO ...

RECANTAR CITANDO

PER 2 ATTORI E UN INTERPRETE INFORMATICO

(VERSIONE TESTUALE)

- RECANTAR CITANDO -

Il pezzo è una riflessione sul linguaggio e sull'interazione tra interpreti e calcolatori, intesi come generatori di rumore (caos) all'interno di un sistema apparentemente condiviso come è quello del linguaggio.

La prima versione prevedeva la presenza di due strumentisti, che utilizzavano il "linguaggio" musicale (non essendo esso fornito di un vocabolario condiviso non può essere linguaggio, ma non è questa la sede per discuterne in termini di semantica) attraverso l'utilizzo discrezionale di una serie di moduli formali connotati organologicamente ed associati arbitrariamente a moduli espressivi il più riconducibili al linguaggio umano verbale.

Lo sviluppo di tale situazione è stato rinviato, preferendo una prima versione incentrata sul linguaggio verbale. Tale scelta, in realtà presuppone una modifica sostanziale dei meccanismi interattivi, dato l'alto livello al quale si svolge il sistema di riconoscimento linguistico.

L'apporto fornito dall'elaboratore è di tipo distruttivo-ricostruttivo, mediante la registrazione dal vivo della recitazione dei due interpreti (attori) , la successiva scomposizione in frammenti e ricomposizione degli stessi.

Affinchè al senso iniziale del testo recitato segua una ricomposizione suscettibile di senso, il testo è stato creato utilizzando parole che contenessero al proprio interno più parole di significato diverso (es. ISOLAMENTO > I, So, La, Isola, Sola, Lame, Amen, Me, Lamento, Mento). Ciò proprio al fine di renderne una ricomposizione dei frammenti in senso plausibile (anche se logicamente scorrelato).

Rispetto alla versione strumentale, dove era maggiormente facile ricostruire "semanticamente" o meglio, più semplicemente in modo musicale, i frammenti degli oggetti sonori registrati, in questo caso ci si trova a lavorare con sillabe e bisillabi che molto difficilmente, una volta ricombinati stocasticamente, possono dare forma a parole di senso compiuto. Senza considerare che se in musica un accento fuori posto, inaspettato, diviene veicolo di un aumento del tasso di informazioni, nel linguaggio verbale, lo spostamento di accenti rende difficilmente comprensibile una parola correttamente formata dal giusto ordine delle sillabe.

Per ovviare a questo tipo di problema si è scelto di ricorrere ad uno stadio di compressione/espansione direttamente in ingresso microfonic, oltrechè la raccomandazione di posizionarsi ad una certa distanza (almeno 20 cm.) dalla capsula microfonica, onde evitare gli effetti di prossimità derivati dall'intercettazione ed esaltazione delle frequenze più gravi.

L'articolazione formale del pezzo tende ad individuare spazialmente i ruoli degli attori interpreti e del regista informatico che presiede la parametrica dell'algorithm di registrazione-elaborazione-riproduzione.

L'attore A è posizionato con il suo microfono ed il suo pedale di attivazione/disattivazione della registrazione all'estrema sinistra della scena, ma in primo piano. Alla sua immediata sinistra, in secondo piano, si trova l'altoparlante "Alter Ego" ovvero il suo "specchio" informatico. Accanto il pedale di switch temporaneo. In maniera simmetrica sono disposti i corrispettivi dell'attore B. (vd. fig. 1)

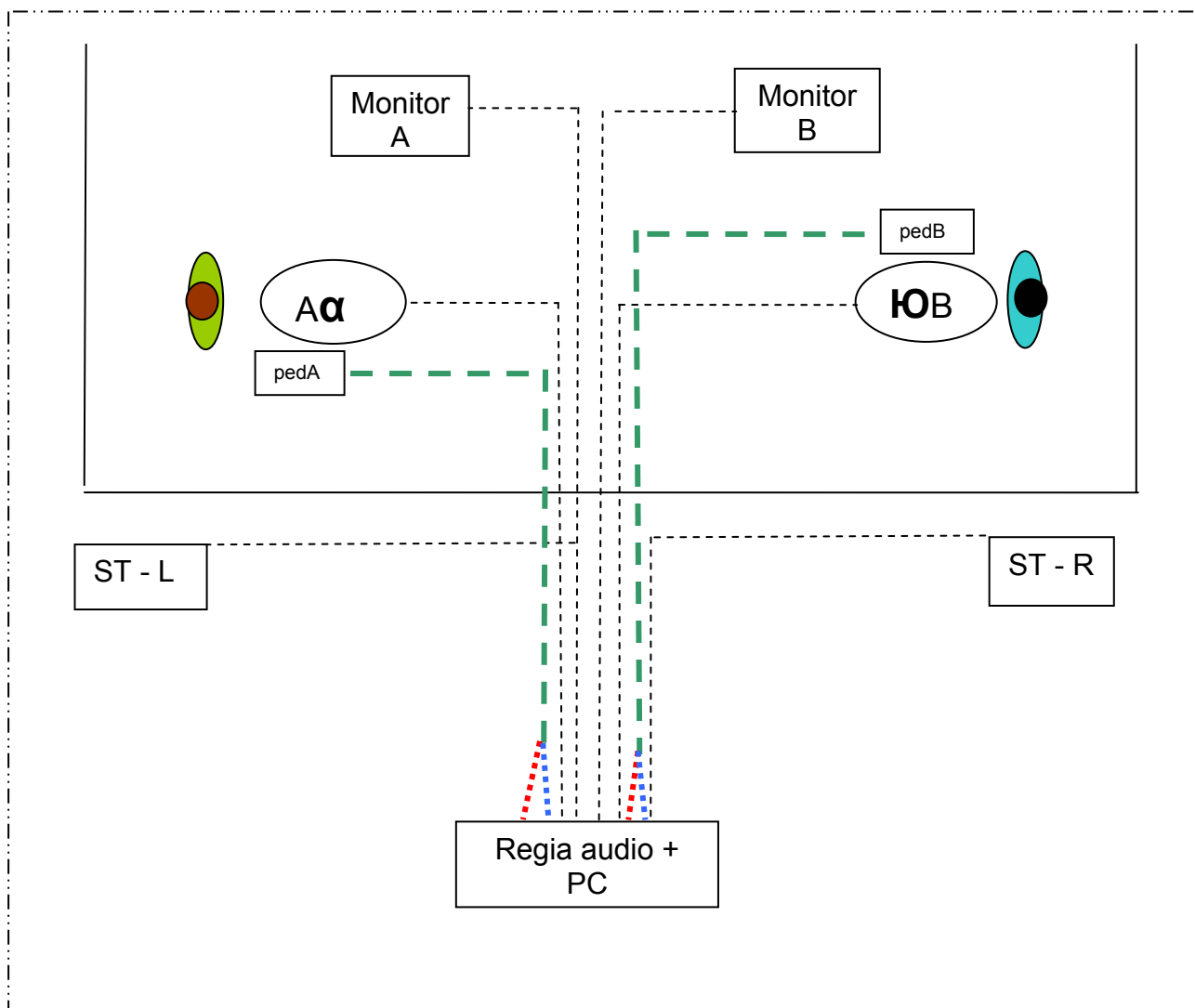


Figura 1

Tale disposizione è motivata dalla possibilità di sovrapposizione/indeterminazione dell'origine del suono.

All'uscita Stereo Principale L sarà inviato sia il segnale diretto dell'ingresso microfonico A che quello elaborato dal PC, similmente per lo Stereo Out R. Invece i monitor sul palco riceveranno solo il segnale elaborato dal PC, costituendo una vera e propria presenza fisica del personaggio speculare/fantasma dell'elaboratore. Con un'opportuna taratura in sede di esecuzione, difficilmente programmabile in anticipo, si provvederà a pesare i segnali elaborati dal PC in modo da inviarne copie differenti in volume e delay ad i monitor e agli altoparlanti di sala. Lo scopo è quello di equilibrare il volume della ripresa diretta microfonica con quella in uscita da due altoparlanti dislocati ad una certa distanza, onde spazializzare il suono come proveniente dallo stesso attore, posto a metà strada dal Main Out L (R) e il mon. A (B).

FORMA

L'intero pezzo è strutturato da quattro distinte scene.

1. **Dialoghi allo specchio** (i due attori separatamente si confrontano con l'elaborazione del proprio segnale registrato in un buffer)
2. **Dialogo diretto** (i due attori dialogano tra loro, fraintendendosi)
3. **Dialogo filtrato** (i due attori dialogano tra loro ma sono filtrati dall'elaboratore che si interpone in vari modi)
4. **Dialogo tra specchi** (i due altoparlanti dialogano tra loro senza la presenza degli attori, utilizzando le registrazioni precedenti e la registrazione reciproca).

1. Dialoghi allo specchio

La prima scena è composta di due parti speculari, ognuna affidata ad un singolo attore.

Questa scena si apre con i soli oggetti presenti sul palco. Il primo attore entra in scena, si avvicina al proprio monitor, lo guarda a lungo poi si posiziona davanti al microfono ed inizia a recitare la prima frase del testo (10" circa) rivolgendosi al proprio monitor A, che, ricordiamo, porta solo il segnale elaborato dal PC.

La prima sezione del pezzo vuole farsi metafora di una moderna deviazione psicologica, quella che vede l'essere umano preferire una "comunicazione" con il proprio IO mediato dall'elaboratore. Mi riferisco all'uso ludico dell'elaboratore, che consente di "misurarsi" con sé stesso scegliendo, a seconda dello stato d'animo, dell'esperienza maturata, della preparazione, diversi e progressivi livelli di difficoltà. Una nuova versione tecnologica del ben noto dialogo allo specchio. La funzione affidata all'elaboratore è quella di rimescolare le sillabe pronunciate dall'attore con velocità e direzioni diverse, alternando pause e ripetendo ciclicamente il processo di tipo randomico. Ogni ciclo verrà interrotto dalla "risposta" dell'attore, che ogni volta che parlerà, premerà un pedale che insieme avvierà la registrazione dei buffer (1 locazione di memoria temporanea ogni volta resettata prima della scrittura) e terminerà la riproduzione ciclica.

La metafora prevede il completo soggiogamento dell'attore da parte della macchina, essa viene rappresentata nel dialogo allo specchio dalla imperturbabilità della macchina, che viene intesa come segno di estrema decisione e forza di carattere (!!) o ancora meglio di ferrea ed indiscutibile coerenza logica. L'attore, catturato fonicamente dalla macchina, si vede restituito a brandelli, con una logica (dichiarata dalla costanza dei tipi di interventi e dalla loro stocastica coerenza) che seppur priva di significato e di apporto significativo di informazioni, non può che affascinare per il narcisistico piacere derivato dal riascoltarsi.

L'attore modificherà il modo di recitazione ogni volta cercando di adeguarsi al ritmo o alla trasposizione più o meno lontana del PC. Questi, comandato dal regista, risponderà all'attore in 8 modi diversi, mediante riproduzioni frammentate con durate diverse e con diverse intonazioni (derivate dalla velocità di riproduzione dei frammenti).

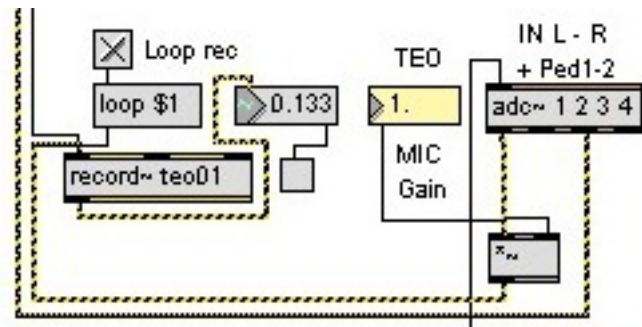
Vediamo ora l'algoritmo del meccanismo di registrazione/elaborazione/riproduzione.

Tutti gli algoritmi utilizzano 4 canali di ingresso e uscita, i primi due dedicati agli ingressi microfonici e alle uscite delle elaborazioni, gli altri due invece servono come *send* e *receive* del segnale di onda quadra inviato e ricevuto dai due pedali per la gestione delle registrazioni / riproduzioni. Questi ultimi ovviamente non sono indirizzati nel mixer per l'ascolto.

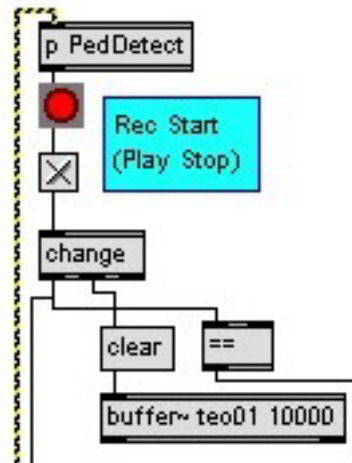
Tale algoritmo è simile a quello utilizzato per il secondo attore, con le⁴

necessarie modifiche dei nomi di buffer , mic input ed output, nonché dei preset.

L'ingresso (ASIO) proveniente dal modulo **adc~ 1** viene moltiplicato per un gain variabile (TEO) ed indirizzato nel modulo di registrazione del buffer **record~ teo01**. Esso presenta un uscita che mostra il progressivo riempimento del buffer segnalandone la fine. Ha un controllo per la registrazione ciclica dello stesso **loop \$1**.



La registrazione del buffer è iniziata ed interrotta dall'azione di un pedale di tipo sustain switch temporaneo, che cioè resta in posizione ON solo durante la pressione. L'implementazione software prevede l'utilizzo di una patch **p PedDetect** per la ricezione di un segnale di controllo (**p square** onda quadra) ed un modulo **change** per la rilevazione del cambio di stato nonché del modulo di invio del segnale, appunto **p square**. Essi sono descritti in seguito. Nella fig. accanto notiamo che il rilevamento dell' on illumina il



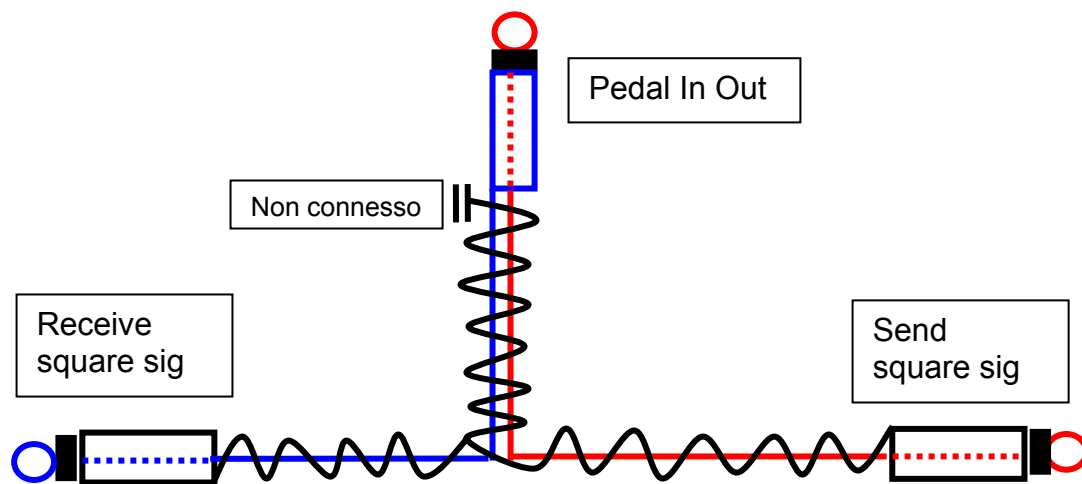
led rosso procurando la selezione della check-box sottostante, che attraverso il modulo **change** avvia la registrazione del buffer, cancella (**clear**) il buffer di 10 secondi **buffer~ teo01 10000** ed effettua una comparazione con 0 con il modulo **==** che serve per l'immediata interruzione della riproduzione.

Lo stesso led si spegnerà al rilascio del pedale, deselegnando la check-box, interrompendo la registrazione ed avviando la riproduzione.

Pedal Detection

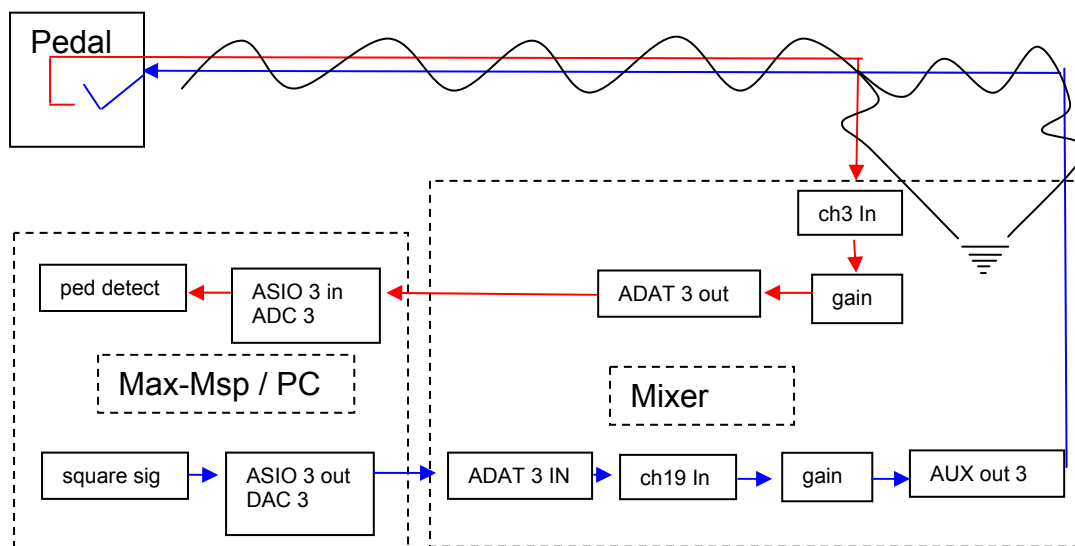
Prima di illustrare il software che implementa l'utilizzo del pedale, è bene soffermarsi un attimo sull'hardware impiegato e le connessioni.

Come abbiamo detto ci si avvale di due pedali simili di tipo sustain (temporary switch). Essi sono connessi e cablati come descritto in figura.



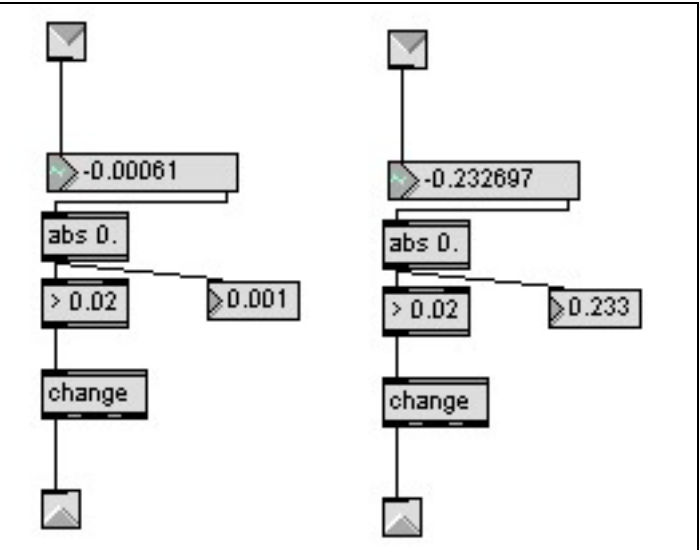
Al pedale viene collegato un jack da ¼ di pollice sbilanciato, al quale è connesso un cavo bilanciato (2 conduttori + maglia di isolamento elettromagnetico). Detto cavo ha un conduttore collegato al tip del jack e l'altro al ground. La maglia invece non è connessa. Gli altri due jack di identico tipo sono invece collegati a due cavi sbilanciati. Ognuno dei conduttori porta al rispettivo tip il segnale variabile, mentre il segnale di riferimento (massa) è connesso ai due ground.

Di seguito il routing audio del segnale relativo.

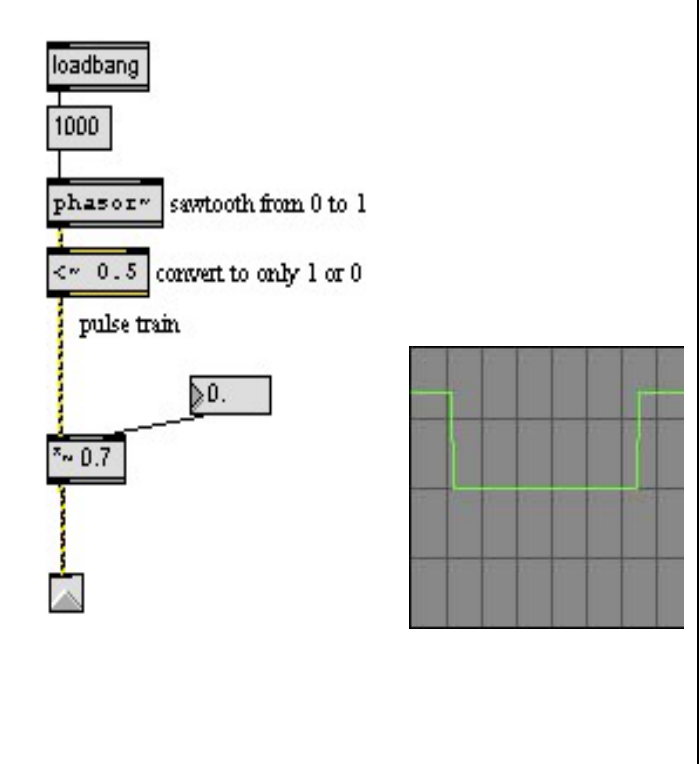


Questa implementazione prevede l'utilizzo del mixer, ma è ovviamente possibile, ma non preferibile, saltare questo stadio se si dispone di una scheda audio con almeno 4 input output analogici. In tal caso la sezione del mixer viene sostituita dall'interfaccia software della scheda audio.

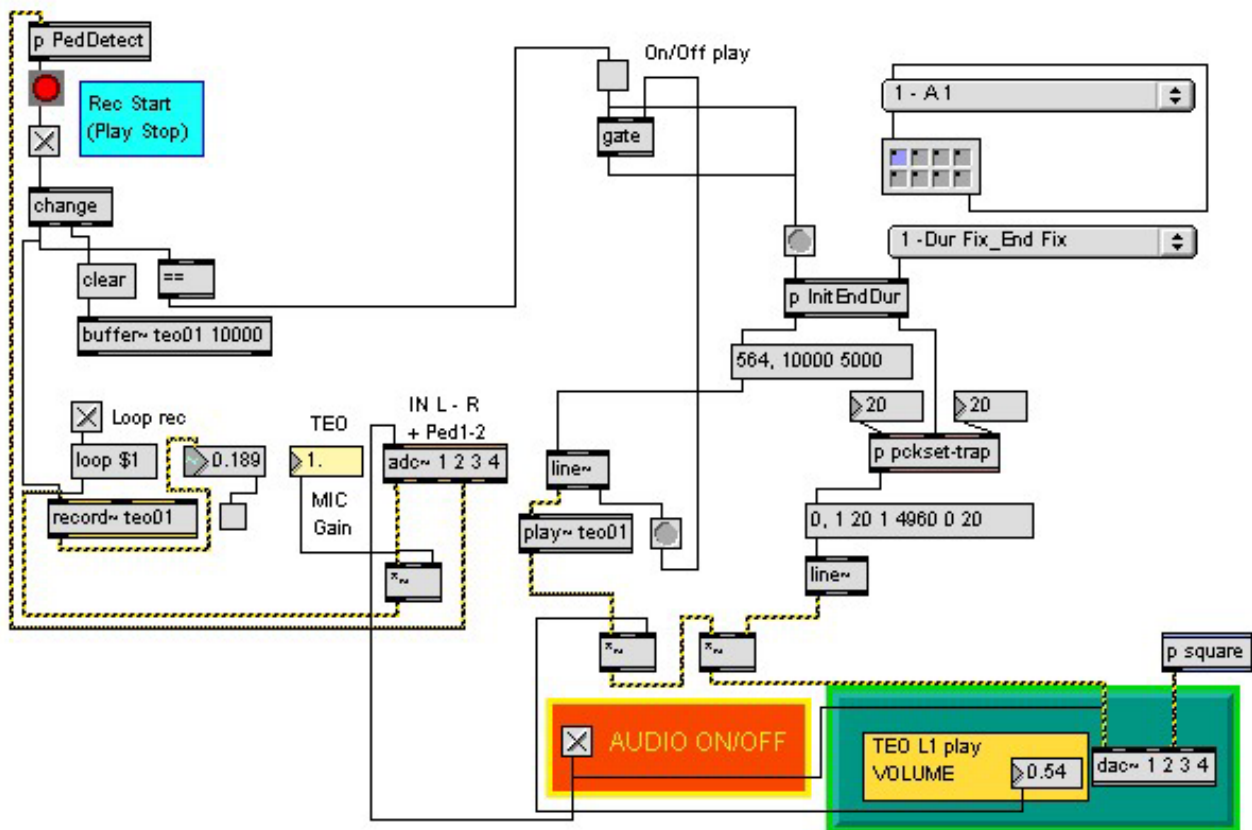
Il modulo **p PedDetect**, a sinistra nella fig. raffigurato in stato Off (pedale sollevato) e alla sua destra in stato On (pedale abbassato), si avvale di un modulo **abs** e di uno di comparazione (>) con una soglia prefissata di intervento (0.02). In cascata il modulo **change** serve per filtrare i dati continui. I valori di ampiezza del segnale variano continuamente e dipendono dagli stadi di amplificazione.



Il modulo che invia il segnale di riferimento per i pedali consiste in un generatore di rampa da 0 ad 1 **phasor~** con frequenza di 1000 Hz convertito da un comparatore audio **<~** che fornisce in uscita un segnale consistente in una sequenza di campioni a zero seguita da una sequenza di uno di uguale lunghezza. Detto segnale viene quindi attenuato opportunamente (*0.7) ed in seguito inviato al DAC3 come precedentemente esposto. Si è scelta la frequenza di 1 khz per evitare che le interferenze causate dalla connessione analogica potessero influenzare significativamente l'ampiezza / voltaggio dello stesso segnale. Il segnale in uscita dal modulo risulta essere una quadra da 0 a 0.7. (Nella fig. a fianco è rappresentata l'uscita del modulo con frequenza di 1 Hz!.)



Ora vediamo l'algoritmo completo, nella sezione sinistra la parte che controlla la registrazione del buffer, in quella destra la parte dedicata alla riproduzione elaborata.



La sezione dedicata alla riproduzione è composta dal modulo **dac~ 1 2 3 4** alle cui uscite sono connessi sia la riproduzione dell'uscita del segnale elaborato, liberamente attenuato (in fig. moltiplicato per 0.54) che il segnale di controllo per il pedale **p square**.

In alto troviamo una check-box titolata "On/Off play", che in realtà si limita, una volta ricevuto il segnale di fine registrazione/avvio riproduzione come precedentemente illustrato dal modulo di comparazione **==**, ad aprire (chiudere se si va in registrazione) il **gate** relativo al ciclo di riproduzione del buffer ottenuto con il modulo **play~ teo01**.

La lettura del buffer è comandata da una **line~**. Detto modulo presenta nell'uscita destra un **bang** che segnala la conclusione della linea stessa. E' questo bang che nel ciclo di riproduzione reinnesca sé stesso.

Il bang in uscita dal gate invece avvia l'esecuzione di un modulo complesso destinato alla gestione dei punti iniziale, finale e la durata di lettura del buffer.

La elaborazione dell'algorithmo consiste infatti in una lettura varia del buffer registrato.

Viste le possibilità di variazione consentite dal modulo (punto d'inizio lettura, punto di fine lettura, durata della lettura, tutti valori espressi in ms) si sono implementate le possibili combinazioni. Da dette combinazioni si è implementata la sub patch **p InitEndDur** che andiamo ad illustrare.

Potendo variare liberamente i tre parametri suddetti, ed utilizzando dei generatori random per la loro variazione, dei nodi di somma e sottrazione, al fine di legare uno dei parametri all'altro, ed escludendo le possibilità di fissare manualmente i punti di inizio di lettura del buffer, cioè avvalendosi per il primo parametro sempre di un generatore random, vengono fuori 9 possibilità, una delle quali esclusa per impossibilità derivate dalla ricorsività. Le possibilità implementate sono le seguenti:

(per "**fissa**" si intende impostata manualmente e fissata per il tempo di esecuzione del ciclo di letture del buffer, fino alla seguente registrazione che l'interrompe; per

"Random" s'intende con una scelta casuale tra una soglia minima e massima impostabili manualmente [offset]; per "**|Inizio - Fine|**" si intende il valore assoluto della differenza tra i due valori di inizio e fine lettura; per "**Inizio+ Durata**" si intende la somma del punto di inizio con la durata)

1. Inizio Random _ Fine fissa _ Durata fissa
2. Inizio Random _ Fine Random _ Durata fissa
3. Inizio Random _ Fine = Inizio + Durata _ Durata fissa
4. Inizio Random _ Fine fissa _ Durata = | Inizio - Fine|
5. Inizio Random _ Fine Random _ Durata = | Inizio - Fine|
6. [*Inizio random _ Fine = Inizio + Durata _ Durata = |Inizio - Fine| esclusa per l'evidente ricorsività intrinseca.*]
7. Inizio Random _ Fine fissa _ Durata Random
8. Inizio Random _ Fine Random _ Durata Random
9. Inizio Random _ Fine = Inizio + Durata _ Durata Random

Dette possibilità sono state inserite negli otto preset comandati dai menu testuali. Di seguito i parametri dei singoli preset:

Preset	RndInitOffset	RndInitMax	END	RndEndOffset	RndEndMax	DUR	RndDurOffset	RndDurMax
1	0	5000	10000			5000		
2	2000	5000	Rnd	5000	5000	2000		
3	0	4000	Init + Dur			2000		
4	0	5000	5000			Init-End		
5	0	4000	Rnd	500	2000	Init-End		
6	0	5000	6000			Rnd	1000	2000
7	0	5000	Rnd	5000	3000	Rnd	1000	2000
8	0	6000	Init + Dur			Rnd	1000	1000

Di seguito (fig. 2) l'algoritmo **p InitEndDur** costruito per l'elaborazione parametrica.

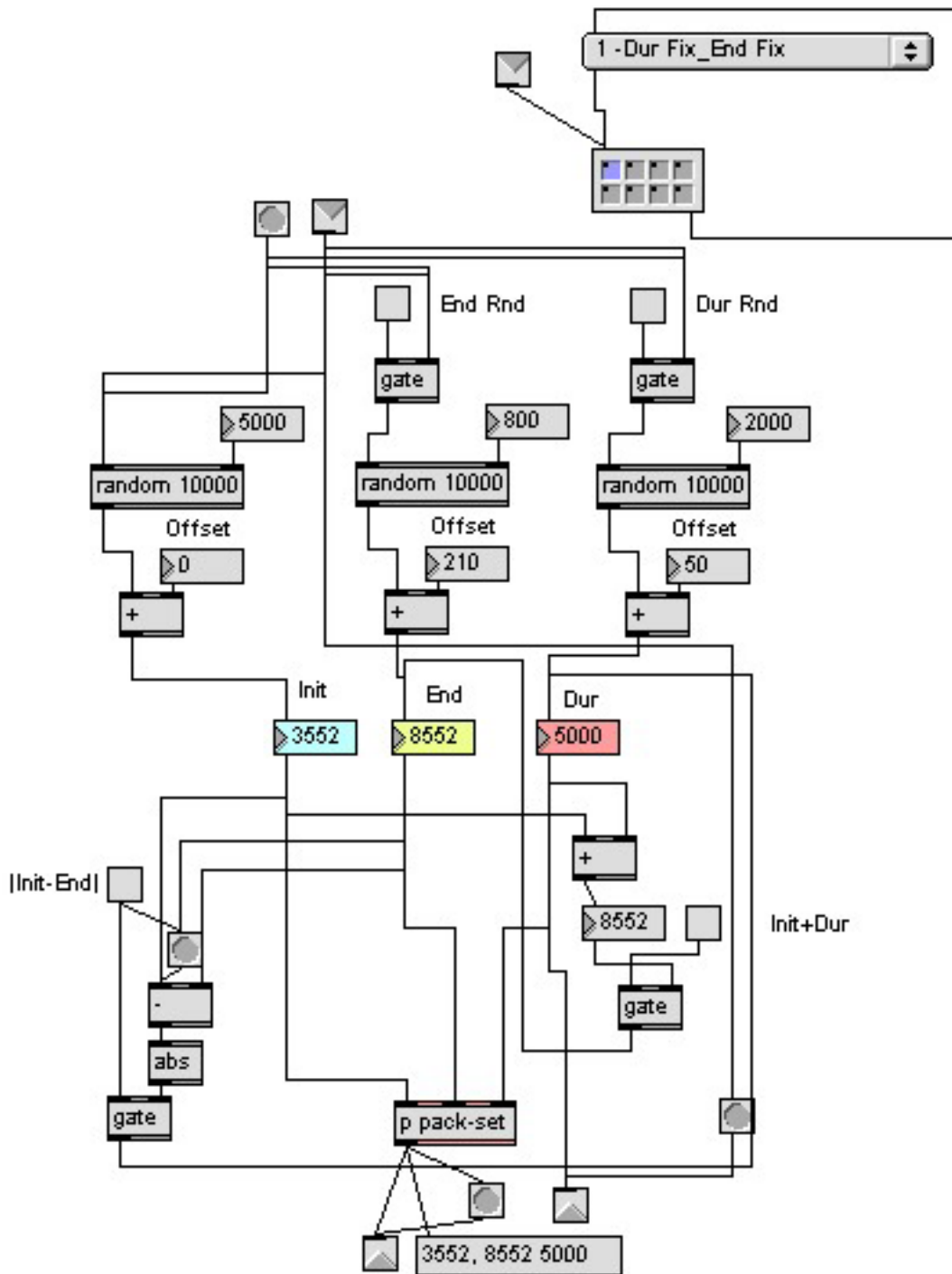


Figura 2

p InitEndDur

Nella parte superiore, dopo i moduli per la gestione dei preset, troviamo i primi due **gate** dei 4 utilizzati per la gestione delle possibilità. **End Rnd** abilita o meno il generatore random per la scelta del punto di fine di lettura del buffer, **Dur Rnd** quello per la durata della lettura. Più in basso, ai due fianchi dell'algoritmo, troviamo gli altri due **gate**. A sinistra **|Init - End|** abilita la differenza assoluta tra il valore d'inizio e la fine di lettura del buffer. A destra **Init+Dur** abilita la somma tra il valore in ms del punto iniziale di lettura e quello della durata.

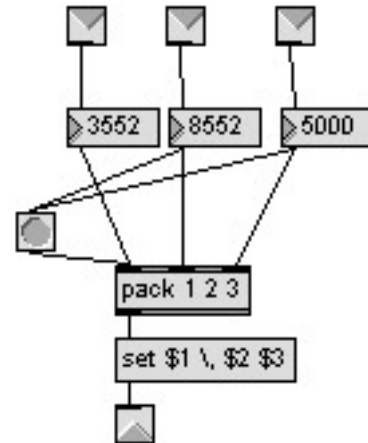
In alto i tre moduli **random 10000** (la dimensione in ms del buffer) forniti di un nodo di somma per l'offset e di un valore massimo (considerare che il valore massimo in realtà è dato dalla somma del valore imposto e dell'offset!).

Al centro, colorati, i tre campi numerici modificabili manualmente, ma in questo caso 10

memorizzati nei preset, dei tempi in ms di inizio, fine e durata di lettura del buffer.

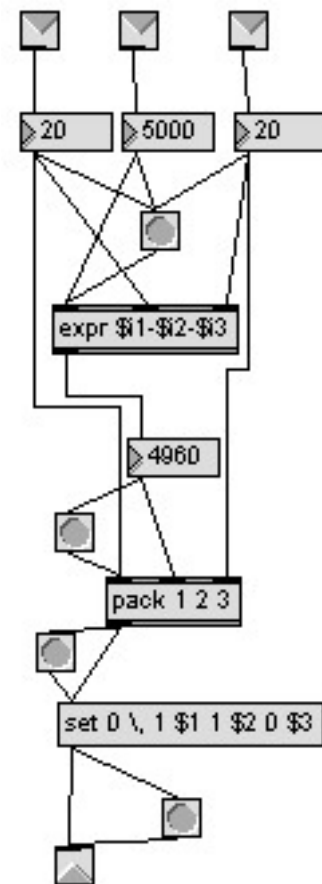
Per finire, in basso troviamo un'altra piccola sub-patch **p pack-set** contenente i moduli necessari all'impacchettamento dei dati da inviare al modulo **line~** dell'algoritmo principale.

Il primo valore non ha bisogno di essere collegato al bang, come gli altri due, essendo esso stesso ricevuto anche come bang.



Il modulo **p InitEndDur** presenta due output, il primo contenente il pacchetto di dati da inviare al **line~** ed il secondo, a destra, solo il valore di durata.

Quest'ultimo servirà ad un'altro modulo **p pcksettrap** (nella figura a destra) che si occupa di involuppare ogni lettura del buffer con un fade in ed un fade out (appunto un trapezoide). Esso prende i valori temporali in ms di attacco e decadimento da due caselle numeriche, questi valori (preimpostati in 20 ms) vengono sottratti dal tempo di durata della lettura del buffer mediante un modulo di espressione matematica **expr \$1-\$2-\$3**, impacchettati dal **pack 1 2 3** ed inviati a loro volta ad un **line~** che moltiplica il segnale prima di essere indirizzato al **DAC 1**.



Il testo

Pur essendo preferibile una completa improvvisazione testuale si sono elaborati dei testi per le scene costruiti in modo da rendere maggiormente probabili ambiguità semantiche.

Si sono ricercate quelle parole che contenessero al proprio interno altre parole di significato dissimile. Quindi si è costruita una tabella contenente le parole divise per numero di sillabe ed infine si sono montate le frasi cercando una plausibilità semantica iniziale. Nella tabella seg., come es., in grassetto le derivate di "isolamento".

In rosso le primitive.

n.	1	2	3	4	5	6	7
1	a	alle	allega	allegata	generalizza	determinazione	generalizzazione
2	al	ama	amare	appiattimento	impoverimento	imprevedibile	indeterminazione
3	con/m	amen	azioni	comprendere	inserimento	organizzazione	
4	da	ani	compone	demeriti	isolamento	sottolineare	
5	de	Appia	comporre	determina	operazioni	stratificazione	
6	di	atti	concerti	editori	prevedibile	stratificazioni	
7	e	Azio	concerto	generali	rettificare		
8	è	bile	diletto	impoverire	sottolinea		
9	fa	certi	disposti	lineare	tecnologico		
10	ha	certo	disposto	negativo	terminazione		
11	i	corre	doveri	organizza			
12	in	dietro	edito	predisposto			
13	la	dito	formato	rettifica			
14	li	dove	genera	rivendere			
15	lo	eri	isola	scendiletto			
16	ma	fallo	lamento	scorrelati			
17	me	falò	legata	scorrelato			
18	ne	fica	linea	stratifica			
19	o	forma	logico	trasformato			
20	per	forse	meriti				
21	pi	gene	merito				
22	re	io	nazione				
23	se	lame	negati				
24	so	lati	occorre				
25	te	lega	opera		segue 2		
26	ti	letto	organi		razio		
27	tra	lizza	poveri		rema		
28	tu	mare	prendere		rende		
29	un	mento	prevedi		retti		
30		meri	rendere		rido		
31		mina	rivedo		rime		
32		mite	rivende		riti		
33		nega	termina		rito		
34		nera	tramite		rive		
35		Nizza	trasforma		scendi		
36		nolo	vendere		scorre		
37		oli			sei (6)		
38		otto			seri		
39		ove			sola		
40		pera			sotto		
41		perla			strati		
42		pia			tori		
43		piatti			uno (1)		
44		pone			vedi		
45		porre			vedo		
46		posti			vende		
47/50		posto			veri		
48/51		prende			zio		

Testi iniziali:

1. Io, predisposto a rivendere l'opera per diletto, vedo un'organizzazione lineare di appiattimento tramite, forse, operazioni generali di impoverimento.
2. la prevedibile stratificazione di un'azione determina l'isolamento di un'opera da comporre e termina con un inserimento negativo e sottolinea un imprevedibile generalizzazione.
3. L'indeterminazione stratifica e trasforma se organizza e compone il concerto formato da editori disposti a comprendere e vendere organi.
4. Tu sei disposto a amare la nazione con me e rendere logico al lamento di un mare sotto forma di isola?

In ulteriore alternativa all'improvvisazione completa del testo, alla lettura dei testi precedenti, si potrà ricorrere ad un'improvvisazione condizionata all'utilizzo dei vocaboli elencati nella precedente tabella.

Ad ogni frase, della durata di non meno di 10 secondi (la registrazione ciclica sovraregistrerà il buffer fino al rilascio del pedale, conservando il resto precedente inalterato), corrisponderà la scelta di uno degli otto preset da parte dell'interprete informatico, per la elaborazione della riproduzione.

Quindi si alterneranno 8 letture/improvvisazioni testuali ed 8 rielaborazioni diverse, il tutto per ciascun attore.

La prima scena "Dialoghi allo specchio" dopo l'esecuzione del primo attore continua con la sua uscita di scena ed il contemporaneo arrivo del secondo attore che si comporta in maniera speculare al primo.

La scena si chiude con il rientro del primo attore.

2.Dialogo diretto (Seconda scena)

I due attori sul palco si guardano finalmente e l'ultimo arrivato si pone davanti al microfono. Quello che ha più capelli sulla testa reciterà la parte A, l'altro la B. Ma non è escluso il contrario.

Le parole tra parentesi indicano il tempo, quelle in corsivo, sempre non dette, chiariscono l'intonazione da dare alla frase troncata.

- [A] (sorridente) Allora ti trovo finalmente ... (*in ottima forma!*)
- [B] (Interrompendo) Come finalmente, non sapevo mi stessi cercando!
- [A] Stavo dicendo che finalmente ...
- [B] (interrompendo) Ma da quanto mi cercavi?
- [A] Ma io non ti stavo proprio cercando, volevo solo dire che finalmente ti trovo, dopo tanto tempo ...
- [B] Addirittura è da tanto tempo che mi cerchi!
- [A] Ma no!, Che hai capito, vuoi farmi finire di parlare ?
- [B] Parla pure, non voglio farti finire di parlare
- [A] ... (3" pausa di comprensione) come non vuoi farmi finire di parlare?
- [B] Quello che ho detto, non intendo farti finire di parlare!
- [A] Come sarebbe a dire, vuoi che io la smetta di parlare?
- [B] Non ho detto questo, ti prego non travisiamo.
- [A] Hai o non hai detto che non vuoi farmi finire di dire quello che stavo dicendo?
- [B] Tutt'altro, ho detto solo che non voglio interromperti, non intendo farti finire di parlare, che significa che voglio farti finire di dire in pace quello che stavi dicendo, insomma voglio che tu continui!
- [A] ... Sarà! ... Comunque io volevo solo dirti che sono contento di trovarti in ottima forma, dopo tanto tempo che mi eri parso decisamente ... mmh ... come dire ... si insomma assolutamente, ... beh non esageriamo ... un po' ...
- [B] (scocciato dall'indecisione) Come mi trovavi prima, grasso, obeso?
- [A] Ma no! ...
- [B] Allora forse invecchiato, malato?
- [A] Ma neanche, non volevo dire questo ...
- [B] Ho capito! Mi trovavi come uno zombie ingrassato!
- [A] Non scherzare, non sei mai stato grasso, è solo per ... (indicando la sua testa)
- [B] ... I capelli, trovi che sono diventato calvo, che sto perdendo i capelli!
- [A] Ma che c'entrano i capelli, quelli prima o poi ... sono come gli autobus.
- [B] Fai presto a dire prima o poi, tu non ne hai perso uno!
- [A] Di capelli?
- [B] No di autobus! Sei riuscito a prendere tutti gli autobus della tua vita, e arrivare dove sei arrivato.
- [A] Ti ringrazio di riconoscermelo!
- [B] Guarda che non era un complimento, volevo dirtelo da un pezzo che mi fai rabbia.
- [A] Ma che dici, e perché?
- [B] Perché hai fatto di tutto ...
- [A] Certo che ce l'ho messa tutta ...
- [B] No, hai "fatto" di tutto, camminando sulle teste di chiunque ti capitava davanti.
- [A] Senti, non sono più tanto felice di averti trovato ... (... *in forma*)
- [B] E allora smettila di cercarmi!
- [A] ... in forma di...
- [B] (interrompendo) Io non ho bisogno di informarmi su nulla!
- [A] ... (pausa) Ma va ...! (gesto con la mano accompagna l'indirizzo osceno)

(I due attori escono dalla scena da parti diverse)

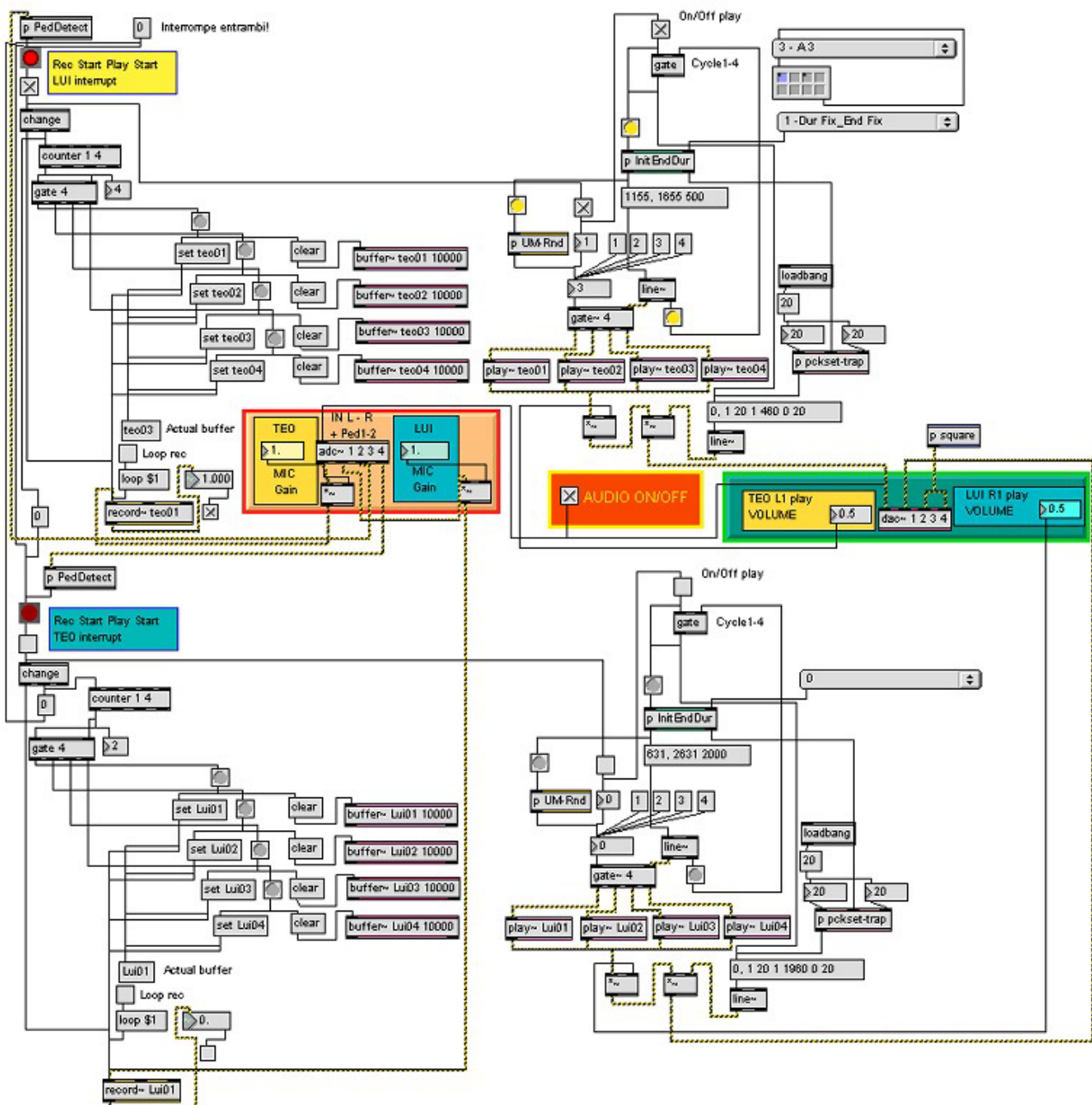
L'interprete informatico ha caricato l'algoritmo successivo.
Dopo pochi secondi rientrano insieme.

3. Dialogo filtrato (terza scena)

La terza parte del brano vuole interporre l'elaborazione informatica delle registrazioni dirette tra i due attori, a modo di filtro/disturbo/rumore. In sostanza si pone come metafora della difficoltà della comunicazione sempre mediata, vuoi dalla tecnologia vuoi dai meccanismi di percezione, vuoi dalle strutture cognitive.

I due microfoni registrano le voci, ma gli attori, pur iniziando a leggere lo stesso testo precedente, vengono costretti, dalle interposizioni/sovrapposizioni/ridondanze elaborative, ad elaborarne via via il contenuto in maniera sempre più caotica e priva di senso, adeguandosi all'assurdità suggerita dalla ricostruzione sillabica del computer.

L' algoritmo si avvale dei moduli di registrazione e riproduzione precedenti, come dell'uso contemporaneo dei due pedali, ma è implementato con le dovute differenze atte alla possibile sovrapposizione delle letture/scritture di 4 buffer per ogni attore.



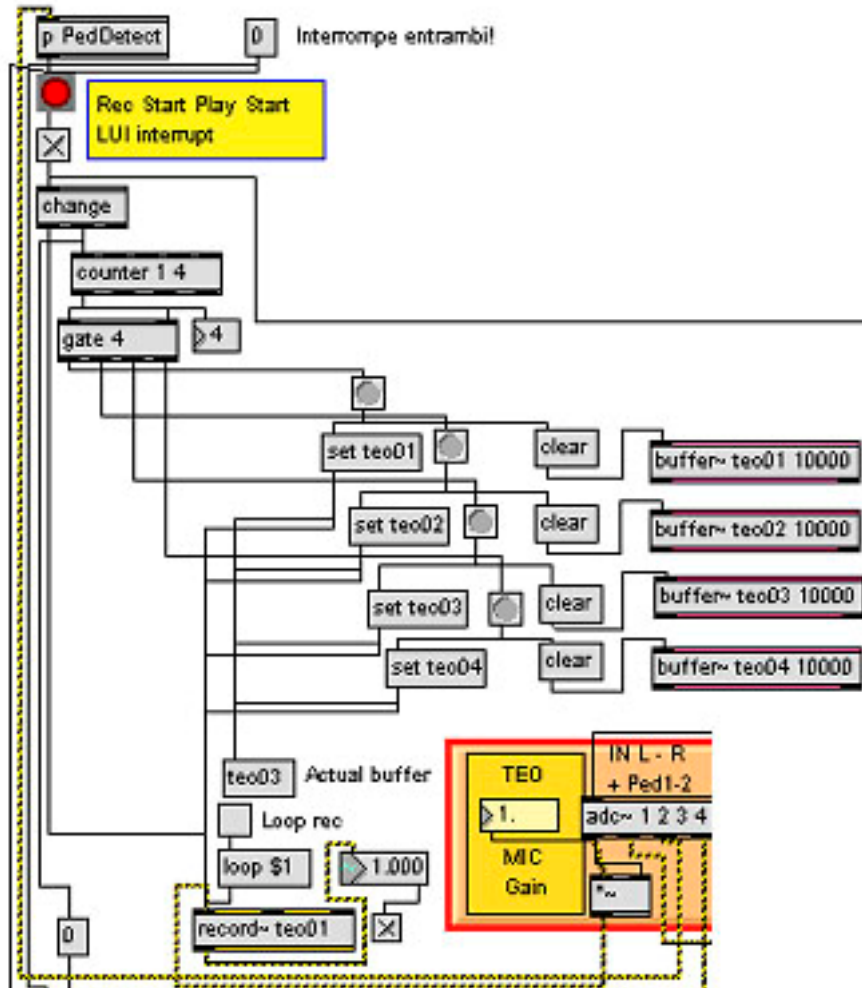
Algoritmo completo per la terza scena. In esso rintracciamo le due sezioni principali disposte verticalmente, nella metà

superiore quella dedicata al primo attore, in quella inferiore l'altro.

In ciascuna sezione individuamo a sinistra la parte dedicata alla registrazione e a destra quella attinente alla riproduzione.

Gli **adc~ 1 2 3 4** ed i **dac~ 1 2 3 4**, con individuali campi numerici per il guadagno degli ingressi microfonici e delle uscite sono disposti al centro dell'algorithmo e contraddistinti da cornici colorate. Al centro l'interruttore principale audio.

Sezione audio IN primo attore

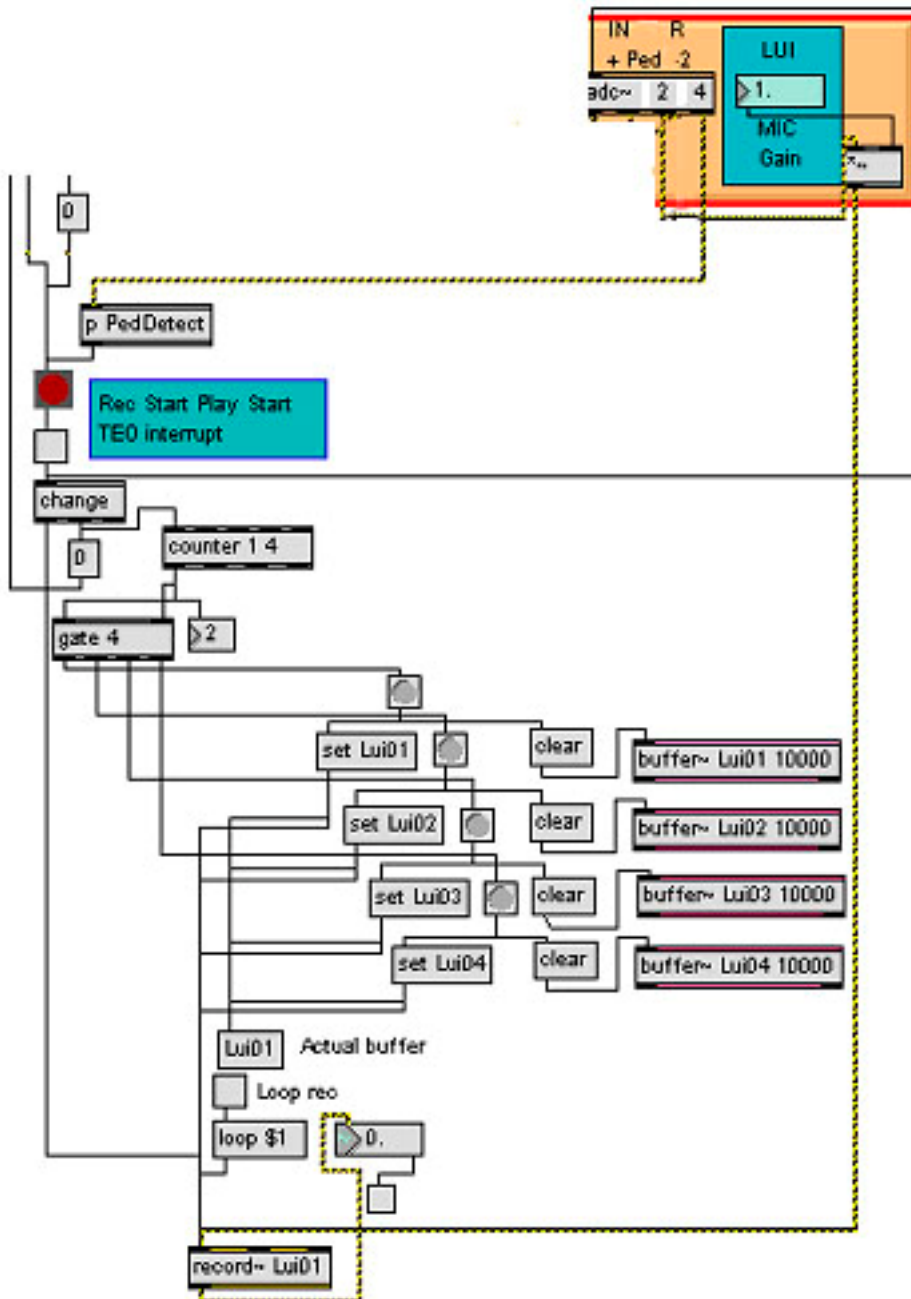


Esaminiamo solo le differenze con i precedenti algoritmi. Abbiamo 4 buffer invece di uno, che comportano l'utilizzo di un **counter 1 4** e di un **gate 4** per la loro rotazione ad ogni nuovo avvio di registrazione. Per il resto la sezione è simile a quella già analizzata.

Detta sezione (chiamata come i buffer relativi TEO, in onore al primo dei due primi interpreti) avrà il proprio analogo nella sezione relativa al secondo esecutore (LUI, Luigi il secondo dei primi interpreti).

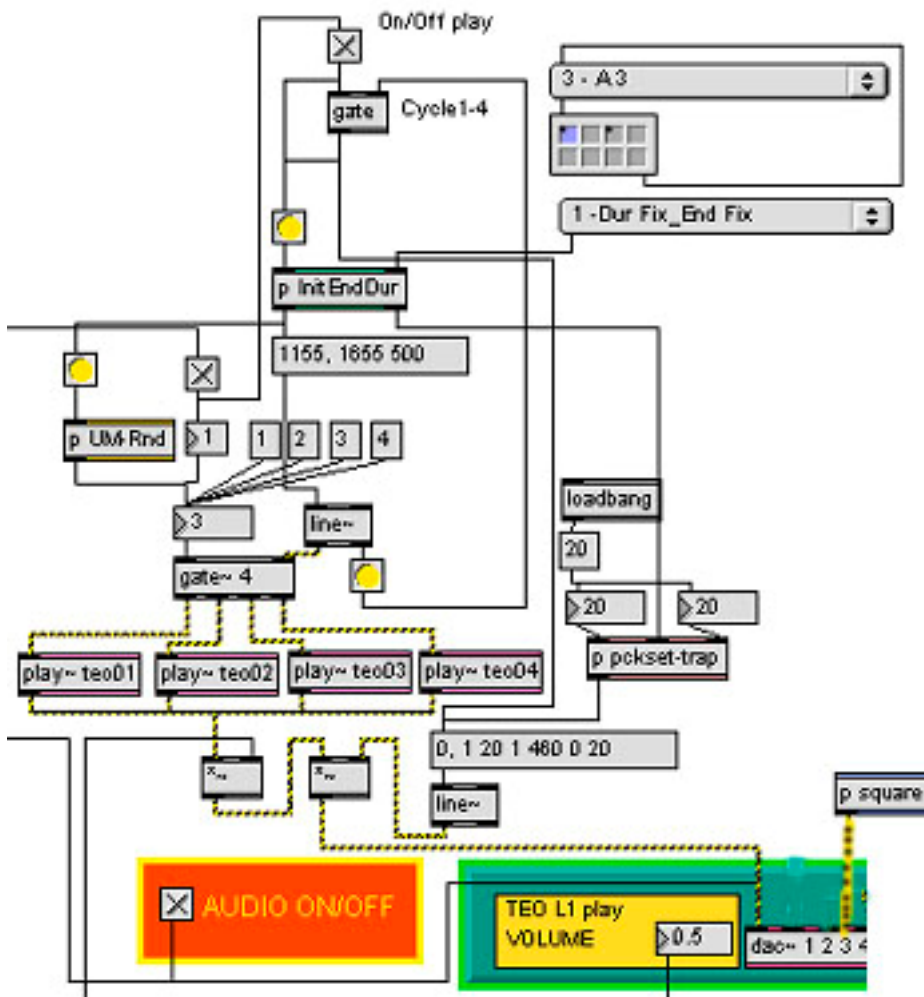
Nei frammenti dell'algorithmo riportati in queste pagine si sono "cancellate" le zone riguardanti altre sezioni.

Sezione audio IN secondo attore



I due interruttori comandati dai pedali in questo algoritmo invece di iniziare/terminare la registrazione e contemporaneamente terminare/iniziare la riproduzione elaborata, hanno funzioni diverse. Infatti ciascuno avvia la propria registrazione e contemporaneamente la propria riproduzione, non solo, ma interrompe la registrazione e la riproduzione dell'altro. Tale critica scelta è realizzata prendendo il secondo output del modulo change, che viene triggerato solo nel cambio di stato da 0 ad 1 (e non viceversa, per filtrare il bang derivato dallo spegnimento [1 ->0], che logicamente avrebbe innescato una incomputabile ricorsività) e triggerando uno "0" che spegne il reciproco interruttore.

La seguente, invece, è la sezione dell' algoritmo riguardante l'elaborazione e la riproduzione.



Anche qui abbiamo i 4 buffer (teo01-04) letti dai moduli **play~**. Questa volta, essendo più d'uno, sono selezionati randomicamente in ciclo seriale dal modulo **p UM-Rnd** (in congiunzione con il **gate~ 4** che ne indirizza la **line~** per la lettura). Di questo modulo parleremo dopo. Al suo fianco destro sono presenti 4 pulsanti per la scelta manuale dei buffer da eseguire, utilizzati in fase di verifica e lasciati come ornamento. Una check-box collegata al meccanismo di registrazione + riproduzione, inizializza il ciclo di riproduzione, con la scelta del primo buffer.

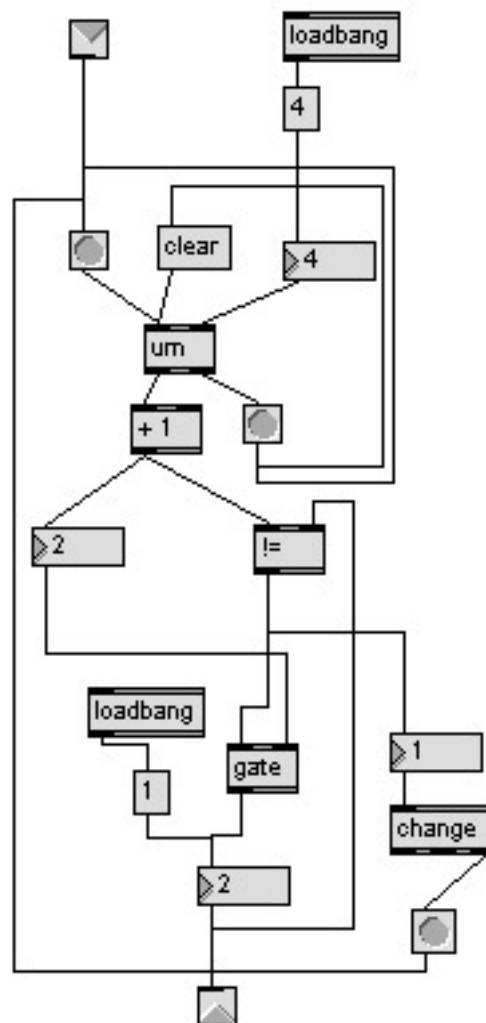
Per quanto invece riguarda la gestione dei frammenti mandati in riproduzione, ci si è avvalsi ancora del modulo **p InitEndDur** appositamente creato e illustrato precedentemente. Come del resto per la "finestratura" degli stessi frammenti, operata come in precedenza dal modulo **p pckset-trap** che applica degli involucri di volta in volta dimensionati al frammento scelto dall'elaboratore stocasticamente. Infine, prima dell'applicazione degli involucri, l'ampiezza del segnale viene controllata utilizzando un campo numerico (TEO L1 play VOLUME).

Il modulo **p UM-Rnd** , raffigurato a destra, mette a disposizione i numeri da 1 a 4 in successione random ma di tipo "seriale" cioè ogni serie di 4 numeri li contiene tutti una volta.

Alla sua uscita è imposto un nodo di somma per elevare il proprio naturale offset da 0 ad 1.

Un modulo di comparazione (diverso da ... **!=**) in congiunzione con **change**, evita che il primo numero della serie sia uguale all'ultimo della serie precedente. In questo caso viene utilizzata la terza uscita di **change**, quella che segnala il passaggio da 1 a 0, che nel nostro caso avviene quando la comparazione di diversità risulta falsa (0). In questi casi un bang ritriggera l'oggetto **um** che fornisce i numeri.

Il comando **clear** imposto all' **um**, resetta la serie ad ogni completamento, segnalato dall'outlet destro di **um**.



I preset sono stati creati secondo una logica che impone un crescendo sia di velocità (riduzione progressiva delle durate) che di complessità. Infatti inizialmente i parametri che governano i range randomici sono stati selezionati in modo da non modificare il "pitch" delle voci ed evitare il loro reverse. Quelli seguenti, man mano riducono la lunghezza delle durate, rendono possibili sporadiche inversioni temporali ed iniziano un modesto *pitch shifting*. Andando avanti le voci diventano sempre meno riconoscibili vuoi per *pitch* che per direzionalità di lettura che per durata minima e maggiore della densità di eventi.

La scena, avviata dalla pressione di uno dei due pedali, si conclude con l'intervento decisionale dell'interprete informatico, dopo aver esaurito le possibilità dei preset.

Sarà in ogni modo possibile aumentarne il loro numero e modificarne la parametrica, ma sempre cercando di conservare un divenire tradizionale dal semplice-laconico-interrotto al complesso-logorroico-ininterrotto. E' di vitale importanza che ciascun attore cerchi di "rispondere", mediante il riutilizzo delle stesse parole in senso *affermativo* (cioè di condivisione della proposizione dell'altro) o *negativo* (contestandolo), alle proposte verbali dell'altro, tenendo il testo a disposizione, od uno di propria o di altrui scrittura comunque basata sull'utilizzo della tabella dei vocaboli, solo come punto di partenza e di "salvezza" in caso di completa afasia creativa.

4.Dialogo tra specchi (quarta ed ultima scena)

La scena finale ha inizio con l'uscita di scena degli attori e la contemporanea attivazione dell'ultimo algoritmo.

Esso utilizza gli ultimi 8 buffer registrati in dei files dall'interprete informatico in più man mano li sostituisce con la registrazione dei frammenti elaborati dal reciproco antagonista (dall'altro canale audio). Quindi l'obiettivo è quello di continuare inizialmente un dialogo "residuale" tra due distinte e ben percepibili fonti (canali destro e sinistro), anche se ormai prive assolutamente di qualsiasi senso intellegibile, per giungere, dopo una serie di processi simili ai precedenti ma molto più radicali, ad un continuum granuloso di reciproco copiarsi, che porterà , con l'aumentare parossistico della velocità (diminuzione delle durate) ad una uniformità pressochè assoluta, un unico ricco suono continuo, metafora di un feedback dovuto allo specchiarsi di specchi.

Progressivamente, quando ormai la prevedibilità della conclusione è totale, si inserirà un riverbero col compito di fondere ulteriormente i suoni.

Un filtro passa-basso chiuderà progressivamente questo rumoroso ed indistinto suono fino a farlo scomparire.

1. Dialogo alternato 4 sec. ciascuno
2. Dialogo alternato 3.5 sec pitch min=0.9 max= 1.1
3. Dialogo alternato dur rnd 2.5~4 sec pitch rnd min=0.7 max=1.4
4. Dialogo alternato dur rnd 2.5~4 sec pitch rnd min=0.7 max=1.4, poss.reverse
5. Riproduzione ciclica contemporanea dei due canali con registrazione reciproca, in modalità append.
6. Riproduzione/registrazione ciclica con minori durate in progressiva ulteriore riduzione.
7. Granulazione riverberata continua delle fonti.
8. Lenta chiusura di un filtro passa-basso.

L'algoritmo preposto utilizza le precedenti configurazioni, con nuove connessioni atte ad alternare o a far coesistere le riproduzioni e le registrazioni reciproche. Sono inseriti i comandi **read** relativi al caricamento dei buffer, sono tolti i comandi ed i moduli relativi ad i pedali, inseriti un riverbero e due filtri passa-basso opportunamente connessi tramite gates ed infine predisposti i preset relativi alle 8 situazioni progressive gestite dall'interprete informatico.

L'algoritmo così modificato si presenta nel seguente modo:

